D.E.L.T.A., Vol. 15, Nº 1, 1999 (27-55)

# Some Notes on Procrastinate and Other Economy Matters*
(Considerações sobre Procrastinar e Outras Questões de Economia)

Jairo Nunes *(Universidade de Campinas)*

*Abstract: This paper argues that Chomsky's (1993) Procrastinate principle is not in consonance with the general guidelines of the Minimalist Program and proposes an alternative account of the preference for covert movement instead of overt movement and the preference for lexical insertion instead of movement. This proposal also accounts for the order of application of certain operations related to deletion of traces.*

*Resumo: Este trabalho argumenta que o princípio Procrastinar de Chomsky (1993) destoa das linhas gerais do Programa Minimalista e propõe uma análise alternativa para a preferência de movimento coberto a movimento aberto e de inserção lexical a movimento. Essa proposta também dá conta da ordem de aplicação de certas operações relacionadas a apagamento de vestígios.*

*Key words: Procrastinate; Derivational Economy; Minimalism; Traces.*

*Palavras-Chave: Procrastinar; Economia Derivacional; Minimalismo; Vestígios.*

## 0. Introduction

One crucial assumption in the Minimalist Program outlined in Chomsky (1993, 1994, 1995) is that the language faculty is a nonredundant and optimal system in the sense that particular phenomena are not overdetermined by linguistic principles and that the linguistic system is subject to economy restrictions specified by Universal Grammar. Part of the Minimalist agenda is thus devoted to investigating the very nature of such economy conditions.

* This paper is a development of section II.10.1 of my dissertation (see Nunes, 1995). The ideas discussed here were presented in courses taught at Pontifícia Universidade Católica do Rio Grande do Sul, Universidade Estadual de Campinas, Universidade Estadual Paulista (Araraquara), Universidade Federal de Alagoas, Universidade Federal do Rio de Janeiro, University of Maryland, and University of Southern California. I am thankful to these audiences. Special thanks to Hans-Martin Gärtner, Max Guimarães, and two anonymous reviewers for helpful comments and suggestions.

This paper discusses Chomsky's (1993) economy principle referred to as *Procrastinate*, according to which covert operations should in principle be preferred to overt ones. I argue that given the overall assumptions of the Program, Procrastinate cannot be taken as a principle of derivational economy and should rather be derived from more general and conceptually sound economy considerations. I explore a suggestion by Chomsky (1995:226) which links derivational cost to the appropriate definition of derivation, showing (i) that the standard effects of Procrastinate can be derived in consonance with core Minimalist guidelines; and (ii) that the fixed order of application of some operations also falls under the same analysis.

The paper is organized as follows. In section 1, I review some of the major features of the Minimalist Program which will be relevant for our purposes. Section 2 presents the motivation for postulating Procrastinate and section 3 discusses whether Procrastinate is in accordance with the general picture sketched in section 1. In section 4, I discuss some attempts to derive the effects of Procrastinate and in section 5, I propose an alternative analysis. Section 6 shows that the proposed analysis receives further support from computations concerning deletion of traces as analyzed by Nunes (1995, 1996, forthcoming). Finally, a brief conclusion is presented in section 7.

## 1. The Minimalist Program: General Picture

Earlier versions of the Principles and Parameters Theory (see Chomsky 1981, 1986, for example) worked with the hypothesis that the linguistic system has several levels of representation encoding systematic information about linguistic expressions. Some of these levels are conceptually necessary, since their output is the input to performance systems which interact with the linguistic system. The Minimalist Program for Linguistic Theory proposed by Chomsky (1993, 1994, 1995:chap. 4) restricts the class of possible linguistic levels of representation to only the ones which are required by conceptual necessity, namely, the ones which interface with performance systems. Under the assumption that these performance systems are the Articulatory-Perceptual System (A-P) and the Conceptual-Intentional System (C-I), the linguistic levels which interface with A-P and C-I are PF (Phonetic Form) and LF (Logical Form), respectively. From the Minimalist perspective, all principles and parameters of the linguistic system should thus be stated in either LF or PF terms, perhaps as modes of interpretation by the performance systems.

Another assumption of the Program is that the language faculty is comprised of a lexicon and a computational system which is strictly derivational (see Chomsky, 1994:5-6, 1995:223-224).[1] The lexicon specifies the items which enter into the computational system and their idiosyncratic properties; the computational system then arranges these items in a way to form a pair (π, λ), where π is a PF object and λ is an LF object. If π and λ are legitimate objects (i.e., they satisfy Full Interpretation in the sense of Chomsky, 1986, 1993), the derivation is said to *converge* at LF and at PF, respectively. If either π or λ does not satisfy Full Interpretation, the derivation is said *to crash* at the relevant level. A derivation is taken to converge only if it converges at both LF and PF.

The pair of legitimate objects (π, λ) must meet the requirement of compatibility. After all, it is not the case that any linguistic sound can be associated with any linguistic meaning. π and λ should thus be based on the same lexical choices. In previous versions of the Principles and Parameters Theory, this compatibility requirement was ensured by D-Structure, which provided the computational system with an array of lexical items structured in a certain way. Under Minimalist assumptions, however, there is no room for a syntactic level such as D-Structure, because it is not an interface level.[2] In order for (π, λ) to be formed according to Minimalist guidelines, it is necessary that the basis for a derivation be an array of lexical items stripped of any substantive property that would make it a syntactic level of representation.

Chomsky (1994:7; 1995:225) proposes that such an array is a *numeration:* a set of pairs (LI, i), where *LI* is a lexical item comprised of (at most) phonological, semantic and formal features, and *i* indicates the number of times that LI is accessed by the operation *Select*. Select pulls out a lexical item from a numeration, reduces its index by one, and makes this lexical item available for further operations of the computational system.

Once the compatibility between π and λ is ensured, one needs to deal with the fact that elements interpretable at the A-P interface are not

[1] For a representational version of the Minimalist Program, see Brody (1995).

[2] For additional conceptual and empirical problems raised by the notion of D-Structure, see Chomsky (1993:sec.3).

interpretable at the C-I interface, and vice versa. At some point in the derivation, the computational system must then split into two parts, one forming $\pi$ and the other forming $\lambda$, which do not interact any further after the bifurcation. S-Structure was the point of this split in pre-Minimalist versions of the Principles and Parameters Theory. The problem from a Minimalist perspective with there being a level feeding PF and LF such as S-Structure is that, since it does not interface with any performance system, it is not conceptually necessary. Thus, every substantive property attributed to S-Structure should be restated within the Minimalist framework in either LF or PF terms.

In the case at hand, the only thing required under Minimalist assumptions is a rule which splits the computation to form the distinct objects $\pi$ and $\lambda$. Chomsky (1993:22) calls this operation *Spell-Out*. Spell-Out is free to apply at any point in a given derivation; "wrong" choices presumably cause the derivation to crash at one of the interface levels.[3] The computation from Spell-Out to PF is referred to as the *phonological component*, the. computation from Spell-Out to LF as the *covert component*, and the computation that obtains before Spell-Out as the *overt syntax*. In addition to containing phonological rules proper, the phonological component includes a morphological subcomponent and also deals with linearization.

Finally, it is assumed that the mapping from a numeration N to $\lambda$, is subject to two conditions (see Chomsky 1994: 8, 1995: 228-229): (i) the *Uniformity Condition*, which states that the operations available in the covert component must be the same as the ones available in overt syntax; and (ii) the *Inclusiveness Condition*, which postulates that $\lambda$ must be built from the features of the lexical items of N.

## 2. Feature Checking, Procrastinate, and Strong Features

Chomsky (1993, 1994, 1995) assumes that lexical and functional heads are already inflected in the numeration. A checking operation made available by overt or covert applications of the operations Merge or Move then allows

[3] The details of the inner workings of Spell-Out have to do with the internal coherence of the system regarding lexical access after Spell-Out, which must be either blocked or very restricted in order to ensure the compatibility between ? and ? See Chomsky (1993:22, 1994:8, 1995:232), Nunes (1995:sec. II.5), and Uriagereka (1997), for different formulations and relevant discussion.

lexical and functional heads to be appropriately paired (if possible). The problem is to show how the parametric variation concerning overt vs. covert movement can be stated without reference to a syntactic level such as S-Structure. This problem basically involves two questions: (i) Why should all languages not have only overt movement? and (ii) Why do some languages have overt movement?

To address the first question, Chomsky (1993:30) proposes an economy principle referred to as *Procrastinate*, which states that covert movement is less costly than overt movement. Regarding the second question, Chomsky (1993:30) proposes that the features of a lexical item may be weak or strong, and that strong features cannot be eliminated in the phonological component.[4] Thus, the only way to prevent strong features from reaching the phonological component is to eliminate them before Spell-Out through the checking operation made available by either Merge or Move. If Merge does not yield a convergent derivation, overt movement is then required.

The fact that overt movement triggered by strong feature checking always violates Procrastinate is not a problem. As an economy principle, Procrastinate only chooses among competing derivations which converge. In the case under discussion, if overt movement does not take place, a strong feature will reach the phonological component and the derivation will crash at PF. Generally put, only the derivations that converge in an optimal way reach the performance systems.

## 3. Problems with Procrastinate

As mentioned in section 1, among the optimality conditions taken to govern the mapping from any given numeration N to its corresponding LF object ?, is the Uniformity Condition, which requires that the operations available in the covert component be the same as the ones available in overt syntax. Let us consider the conceptual motivation behind the Uniformity Condition for a moment.

[4] For alternative views of strong features, see Chomsky (1994:9, 1995:232-235), Nunes (1995:sec. II.6.2), and Uriagereka forthcoming.

If there were operations that by definition could only apply before or after Spell-Out, objects resulting from improper applications of "overt operations" in the covert component should be filtered out by LF and objects resulting from improper applications of "covert operations" in overt syntax should be ruled out by Spell-Out. This, however, would render Spell-Out a syntactic level of linguistic representation, going against the Minimalist goal to eliminate non-interface levels (see section 1). By requiring that the same operations be available for covert and overt computations, the Uniformity Condition renders LF able to filter out illicit objects which would otherwise have to be ruled out by Spell-Out; in turn, this has the effect of stripping Spell-Out of a substantive property which would make it a level of representation.

Bearing these considerations in mind, let us examine the Procrastinate principle introduced in section 2. Suppose for the sake of the argument that in a given convergent derivation, Move has applied overtly in absence of strong features. Obviously, the result of this undesirable movement cannot be ruled out at LF, given that the derivation converges; apparently, the only way to prevent this case is to postulate that Spell-Out rules out the output of overt movement in absence of strong features, which would in turn render Spell-Out a level of representation. Therefore, Procrastinate, as formulated in Chomsky (1993), does not fit well in the system. Postulating an inherent difference between overt and covert movement operations amounts to saying that these are two different types of operation, violating the Uniformity Condition and requiring a non-interface level to deal with one of them.

The general assumptions of Minimalist Program thus lead us to the conclusion that, even if Procrastinate is empirically accurate, it should not be taken as a principle of derivational economy. Rather, it should be simply taken as a description of the results of more abstract economy computations. In the following sections, I discuss some proposals in the literature which attempt to derive the effects of Procrastinate, and advance a new alternative.

## 4. Some Attempts to Derive the Effects of Procrastinate

### 4.1. Nunes (1994) and Kitahara(1995)

Assuming the copy theory of movement, according to which a moved element leaves behind a copy which gets deleted in the phonological component (see Chomsky, 1993:35), Nunes, 1994 and Kitahara,1995) attempt to derive the effects of Procrastinate from economy considerations regarding the number of operations required by overt movement. Leaving aside the technical details and differences between these two papers, their point is that overt movement entails extra work in the phonological component. The derivation underlying the sentence in (1a) below, for instance, with overt object movement for purposes of Case checking requires one application of deletion to eliminate the lower copy of Mary and should thus be more costly than the one in (2b), which requires no such operation. In other words, covert movement should always be the preferred option; overt movement should only be employed if the derivation does not converge otherwise, as in instances involving strong feature checking.

(1) a. *John Mary saw.
b. [$_{\text{AgrsP}}$ John [$_{\text{TP}}$ T [$_{\text{AgroP}}$ Mary [$_{\text{Agro'}}$ Agro [$_{\text{VP}}$ John saw Mary ] ] ] ]
(2) a. John saw Mary.
b. [$_{\text{AgrsP}}$ John [$_{\text{TP}}$ T [AgroP Agro [$_{\text{VP}}$ John saw Mary ] ] ] ]

The virtue of the proposals by Nunes (1994) and Kitahara (1995) is that they rely upon general economy considerations regarding the number of applications of the operations of the computational system in a given derivation, without ascribing inherent cost to overt operations. Therefore, the effects of Procrastinate are derived in compliance with the Uniformity Condition on the mapping from N to λ.

The problem with these proposals, however, is that they resort to global computations. Chomsky (1995:chap. 4) has argued, based on conceptual as well as empirical grounds, that economy comparison should not involve global computations. The (convergent) derivations of (3a) and (4a) below, for instance, each involve one overt movement operation and hence, one violation of Procrastinate. If violations of Procrastinate are to be counted in a global fashion, the derivations of (3a) and (4a) should then be equally economical and pattern alike, contrary to fact.

(3) a. There don't seem to be men in the list.
b. [ there$_i$ don't seem [ t$_i$ to be men in the list ] ]

(4) a. *There don't seem men to be in the list.
b. *[ there don't seem [ men$_i$ to be t$_i$ in the list ] ]

In order to account for the contrast between (3a) and (4a), Chomsky (1995:346) proposes that economy should be computed at every step of the derivation. Consider the step after the computational system has assembled (in a cyclic fashion) the structure in (5) below, whose T head has a strong feature requiring the subject position to be filled. Insertion of *there* to check the strong feature, as in (3b), is more economical than movement of *men*, as in (4b), because the latter violates Procrastinate; hence the contrast between (3a) and (4a).

(5) [ to be men in the list ]

Let us return to the discussion of the proposals by Nunes (1994) and Kitahara (1995). According to these proposals, the decision of whether or not the object should move overtly after the structure in (6) below is assembled in the derivation of (1a) or (2a) depends on later computations in the phonological component, after the full AgrsP is assembled and Spell-Out has applied. However, if economy can be determined based on such global computations, one cannot account for the contrast between (3a) and (4a), as argued by Chomsky (1995). Thus, a uniform account of the data in (1)-(4) is still to be provided.

(6) [$_{AgroP}$ Agro [$_{VP}$ John saw Mary ] ]

4.2. Chomsky (1995)

A more promising approach can be found in Chomsky's (1995:sec. 4.4.4) theory of movement. Chomsky (1995:262-263) observes that if movement operations are triggered by feature checking, Minimalist considerations would lead us to expect Move to operate with features, rather than categories. Chomsky (1995:262-263) then proposes that the operation Move does target features; however, properties of the phonological component require that when a feature of a lexical item or a phrase moves, all the other features of that category be pied-piped. Morphology presumably is not able to operate with isolated features or other scattered parts of words. Thus, overt movement of a feature F has the appearance of movement of a category containing F; on the other hand, since covert movement does not feed Morphology, it need not (therefore must not) resort to generalized pied-piping. Movement of a given feature F for checking purposes is therefore subject to the condition in (7) (see Chomsky, 1995:262):

(7) F carries along just enough material for convergence.

Crediting H. Kitahara and H. Lasnik by the observation, Chomsky (1995:264) notes that "the proposed economy principle provides a further rationale for the principle Procrastinate: nothing at all is the least that can be carried along for convergence and that is possible only if raising is covert, not entering the phonological component".[5]

This suggestion also has the merit of deriving some aspects of Procrastinate without assigning inherent cost to overt operations, thereby satisfying the Uniformity Condition. However, it appears to be restricted to the choice between overt and covert movement in absence of strong feature checking and does not extend to the choice between lexical insertion and overt movement for purposes of strong feature checking, which also falls under Procrastinate in Chomsky's (1995:chap. 4) system. Recall that a convergent derivation involving the step in (5) must check the strong feature of T. It is plausible to assume that the morphological restrictions regarding scattered features mentioned above also exclude a derivation in which only

[5] Chomsky (1995:265) assumes that "Move F automatically carries along FF(LI), the set of formal features of LI". However, the pied-piping of the remaining formal features of LI when a feature F of LI is moved in the covert component in principle should also be excluded by the economy condition in (7). If true, this apparent departure of optimality needs to be accounted for. It could be the case that the Move operation just happens to deal with sets of features or sets of sets of features, but not with single features. Another possibility to consider is that movement in the covert component may actually target heads, which only have formal and semantic features after Spell-Out. Yet another possibility is that the derivational cost with respect to feature movement may take into consideration three variables: number of features moved, number of applications of Move, and number of checking relations made available by the moved features. The idea is that the most economical derivational step is the one which allows the largest number of checking relations with fewest number of features in a single application of Move. I will leave the choice among these three options pending on further research.

the categorial feature of *there* in (3b), for instance, merges with the structure in (5) to check the strong feature; rather, the whole category *there* (including phonological features) must merge with (5). Once "generalized pied-piping" is arguably required for overt applications of both Merge and Move, one needs to resort to an independent economy criterion to choose between the derivations in (3b) and (4b). Again, we are still in need of a uniform account of (1)-(4).

4.3. Groat and O'Neil (1996)

Groat and O'Neil (1996) propose an alternative model to the one laid out in Chomsky (1993), as far as the notions of Spell-Out and movement are concerned:

> In our model, a derivation proceeds until all features, weak and strong, have been checked, yielding a single "final" phrase-marker $K_f$, which is the object of the interpretive mechanism and of the phonological component. In other words, Spell-Out and LF interpretation take the same $K_f$ as their input. All syntactic operations have taken place before interpretation and before PF; there are no post-Spell-Out syntactic operations. (Groat and O'Neil, 1996:124)

In this system, the difference between "overt" and "covert" movement is expressed in terms of whether the head or the tail of the chain is phonetically realized, which should take place in compliance with the principle in (8a) and the economy condition in (8b) (Groat and O'Neil's (7)).

(8) a. Strong features may be checked only in a checking relation with node specified for phonological features.
  b. Moving phonological features to the head of the chain is more costly than leaving them in the tail of the chain.

The effects of Procrastinate regarding the preference for "covert" instead of "overt" movement appear to be derived in a natural fashion in this system, given that more features are moved when movement takes place overtly. However, this result is achieved with a substantial complication of the inner workings of the movement operation. In Groat and O'Neil's (1996:125) own words, "forming a chain results in copying all syntactic features of the category moved, but does not copy the category's phonological matrix: it either moves it to the new position or fails to move it". That this complication is not without problems can be illustrated by two facts.

First, if the only difference between overt and covert movement is phonetic realization (movement of phonological features), we should expect the Portuguese sentences in (9) and (10) with and without overt wh-movement, respectively, to have the same possibilities of interpretation for the reflexive, which is not the case.

(9) Que fotografia de [ si mesmo ]$_{i,j}$ Pedro$_j$ disse que Paulo$_i$ viu?
which picture of self own Pedro said that Paulo saw
'Which picture of himself did Pedro say that Paulo saw?'

(10) Pedro$_j$ disse que Paulo$_i$ viu que fotografia de [ si mesmo ]$_{i/*j}$ ?
Pedro said that Paulo saw which picture of self own
'Which picture of himself did Pedro say that Paulo saw?'

The other potential problem for this approach is posed by constructions involving wanna-contraction, such as (12) below, for instance (see Nunes, 1995:sec. III.4.3.2). Given the sentence in (lla), where the second instance of *who* precedes *to*, it is safe to assume that a strong feature is checked in the embedded subject position; hence, (12a) should involve (at least) two instances of movement, as represented in (12b):

(11) a. Who wants who to win the prize?
b. [$_{CP}$ who wants [$_{IP}$ who$_i$ to [$_{VP}$ t$_i$ win the prize ] ] ]

(12) a. Who do you want to/*wanna win the prize?
b.[$_{CP}$ who$_i$ do you want [$_{IP}$ t$_i$ to [$_{VP}$ t$_i$ win the prize ] ] ]

According to Groat and O'Neil's analysis, the phonological features of *who* in (12b) should be moved to the embedded subject position to check a strong feature and then moved to the matrix Spec of CP to check another strong feature. Given that there are no phonological features intervening between *want* and *to*, we should then expect contraction to be allowed, contrary to fact. Notice that we cannot ascribe the impossibility of (12b) to the intervening formal or semantic features of the intermediate trace of *who*. Were that the case, the formal or semantic features of PRO in (13b) should also block contraction, again contrary to fact[6]:

(13) a. I want to/ wanna win the prize.
b. I want $[_{CP}$ PRO$_i$ to $[_{VP}$ t$_i$ win the prize] ]

It might be the case that these problems can be solved if Groat and O'Neil's proposal is recast in terms of Chomsky's (1995) Move-F approach, a possibility that I will not explore in this paper. The point to be borne in mind here is that even if the problems pointed out above are overcome and the effects of Procrastinate regarding covert vs. overt movement can be derived along the lines Groat and O'Neil suggest, we would still need an independent economy criterion to choose between Merge and Move, as in Chomsky's Move-F analysis (see section 4.2).

## 4.4. Kitahara (1997)

Adopting Chomsky's (1995:chap. 4) general proposal that covert movement involves only sets of formal features, whereas overt movement involves whole categories, Kitahara (1997:chap. 2) attempts to derive the effects of Procrastinate through a global economy condition minimizing the number of applications of what he calls *elementary* operations:

(14) *Shortest Derivation Condition:*
Minimize the number of elementary operations necessary for convergence.

Kitahara proposes that Merge and Move should be decomposed into the more basic operations of *concatenation and replacement*: cyclic applications of Merge or Move involve only concatenation, whereas noncyclic ones involve concatenation and replacement.[7] In addition, assuming that the

[6] As Groat and O'Neil (19%:fn. 3) acknowledge, it is also not immediately obvious in their system how to account for the fact that strong features can apparently be checked by elements without phonological features, such as PRO and null operators.

[7] Replacement is to be understood in the context of Chomsky's (1995:chap. 4) phrase structure building algorithm, according to which given a structure Σ: with constituents α and K *noncyclic* movement of α to target K concatenates α and K forming the object L, and *replaces* K by L in Σ, yielding the new structure Σ'

phonological and semantic features of a lexical item (PF(LI) and SF(LI), respectively) "are interpreted *only once*" (p. 35), Kitahara also proposes that overt movement necessarily induces covert erasure of SF(LI) of LI or of its trace, where erasure is taken to be an application of replacement substituting an empty element ∅?for SF(LI).[8]

With these assumptions, Kitahara successfully derives the effects of Procrastinate regarding verb movement in languages like English, as illustrated in (15), and the preference for Merge instead of Move in the case of (3) and (4), repeated below in (16) and (17):

(15) a. John often sees Mary.
b. *John sees often Mary.

(16) a. There don't seem to be men in the list.
b. [ there$_i$ don't seem [ t$_i$ to be men in the list ] ]

(17) a. *There don't seem men to be in the list.
b. *[ there don't seem [ men$_i$ to be t$_i$ in the list ] ]

Covert verb movement in (15a) involves the application of two elementary operations: concatenation of the formal features of the verb with T, and the replacement of the resulting syntactic object in the larger structure. The derivation of (15b), on the other, requires the application of the same operations plus an additional replacement operation to erase SF(LI) of one of the links of the verb chain. The Shortest Derivation Condition in (14) selects the more parsimonious derivation between these two, yielding the contrast in (15). The contrast between (16) and (17) is accounted for in a similar manner. Overt movement of *men* in (17) triggers erasure of SF(LI) of one of the links of the NP chain, but movement of *there* in (16) does not, since expletives arguably have no semantic features; hence, (16) blocks (17).

When other constructions are considered, Kitahara's approach faces

[8] ∅ is taken to be "an actual symbol of mental representation with no feature" (Kitahara,1997:34). Putting aside the dubious nature of such contentless element, the introduction of a symbol which is not part of the initial numeration in the course of the derivation is at odds with the Inclusiveness Condition, according to which LF objects are built from the features of the lexical items of the initial numeration (see section 1).

some problems which compromise the analysis as a whole. For instance, in this system overt and covert object movement for purposes of Case checking have the same cost, because both options involve two elementary operations: if object shift takes place overtly, it involves concatenation of the object and the vP (a projection of a light verb) and the erasure of SF(LI) of one link of the object chain; if it takes place covertly, it involves the concatenation of the formal features of the object with the relevant head and the replacement of the resulting syntactic object in the large structure. Although this may be a welcome result regarding the optionality of object shift in Icelandic, it is certainly an undesirable result for languages such as French and English.

In order to account for the absence of object shift in English-type languages, which lack verb movement, Kitahara resorts to Chomsky's (1993) Minimal Link Condition (MLC) in terms of equidistance, according to which object shift requires verb movement (past the subject).[9] Since there is no overt verb movement to T in English, the computational system chooses the derivation without object shift, which does not violate the MLC. This line of reasoning however cannot be extended to languages such as French, where overt verb movement should render covert and overt object shift equally costly. As Kitahara (p. 114:fn. 26) acknowledges, an additional parameter would be necessary to make the distinction between Icelandic and French.

It should also be pointed out that by assuming both the version of the MLC in terms of equidistance proposed in Chomsky (1993) and the clausal structure without Agr projections assumed in Chomsky (1995:chap. 4), Kitahara's analysis overgenerates. For instance, it wrongly rules in the derivation sketched in (18) in a language with overt verb movement to T:

[9] The relevant definitions for the following discussion of locality of movement are given in (i)-(iv) below (see Chomsky, 1993:11-19 for original formulation, and Nunes, 1995:sec. II.7, Nunes and Thompson, forthcoming: sec. 8, and Uriagereka, forthcoming for discussion).

(i) *Max(α):*

The least full-category maximal projection dominating α

(ii)Domain of α (δ (α)):

The set of categories contained in Max(α) that are distinct from and do not contain α

(iii)*Minimal Domain of* α *(Min(δ(α)):*

The smallest subset K of (α) such that for any γ ∈ δ(α), some β ∈ K reflexively dominates γ

(iv)*Equidistance:*

Where α and β are targets of movement for a category γ, if α and β are in the same minimal domain, they are equidistant from γ

(18) a. [$_{vP}$ SU-acc [$_{v'}$ V+v [$_{VP}$ t$_{V}$ OB-nom ] ] ]
b. [$_{vP}$ SU-acc [$_{v'}$ t$_{SU\text{-}acc}$ [$_{v'}$ V+v [$_{VP}$ t$_{V}$ OB-nom ] ] ] ]
c. [$_{TP}$ V+v+T [$_{vP}$ SU-acc [$_{v'}$ t$_{SU\text{-}acc}$ [$_{v'}$ t$_{V+v}$ [$_{VP}$ t$_{V}$ OB-nom ] ] ] ] ]
d. [$_{TP}$ OB-nom [$_{TP}$ V+v+T [$_{vP}$ SU-acc [$_{v'}$ t$_{SU\text{-}acc}$ [$_{v'}$ t$_{V+v}$ [$_{VP}$ t$_{V}$ t$_{OB\text{-}nom}$ ] ] ] ] ] ]

In (18b) an accusative subject moves to the outer Spec of vP to check its Case against the verbal complex V+v, and in (18c) the verbal complex moves to T. Crucially, after the verb movement in (18c), the two Specs of vP and Spec of TP fall under the minimal domain of the chain (V+v, t$_{V+v}$) and are, therefore, equidistant from the object (see fn. 9); the nominative object is then allowed to move to Spec of TP to check its Case in compliance with the MLC, yielding (18d). Thus, Kitahara's assumptions lead to the wrong prediction in languages with overt movement to T, a surface sequence corresponding to "John-nom kissed Mary-acc" would be ambiguous between the interpretations 'John kissed Mary' and 'Mary kissed John'.

Given the overall adjustments that are required for Kitahara's system to derive the timing of object shift in different languages without resorting to Procrastinate, it is not clear whether it is a better alternative to Chomsky's 1995:chap. 4 system, which dispenses with the notion of equidistance and assigns an optional strong feature to the light verb in languages such as Icelandic. If these problems are not overcome, the partial derivation of the effects of Procrastinate regarding (15)-(17) becomes substantially weakened.

## 5. An Alternative Approach

Based on a suggestion by Chomsky (1995: chap. 4) about the conceptual grounds for the postulation of economy conditions, below I explore an alternative approach which provides a unified account of the preference for covert movement instead of overt movement and the preference for lexical insertion instead of movement.

### 5.1. Derivational Cost of the Operations of the Computational System

Chomsky (1995:226) suggests that the computation of derivational cost hinges on whether an operation is a defining property of derivations or whether it is associated with a convergence condition on derivations. For Chomsky

(1995:225-226), a derivation is a sequence of symbolic elements S mapped from a numeration N such that the last member of S is a pair (π, λ) and N is reduced to zero (that is, for any Ll of N, i = 0). A given derivation is said to be cancelled if an illegitimate operation is performed during the computation, if the pair (π, λ) is not formed, or if the numeration is not exhausted (see Chomsky, 1995:225-226).

If the applications of Select, for instance, are insufficient to exhaust the numeration, the derivation is cancelled and no questions of convergence or economy arise. Similar considerations hold of the operation Merge, which takes two syntactic objects and replaces them with a single object. Assuming that it is a defining property of a derivation that λ is formed from a single syntactic object, the computational system must then employ sufficient applications of Merge.[10] If such a requirement is not met, the derivation is cancelled and no questions of convergence or economy can be raised.

The operations Move, Delete, and Erase, on the other hand, are associated with convergence conditions. If they do not apply, a derivation may eventually be formed, but at least one object of the pair (π, λ) violates Full Interpretation. Chomsky then suggests that the operations Move, Delete, and Erase, which are required for the pair (π, λ) to be legitimate and interpreted by the performance systems are derivationally costly, whereas the operations Select and Merge, which define what is a possible derivation, have no derivational cost.

Let us now see how this conceptual basis for the computation of derivational economy allows us to derive the effects of Procrastinate.

## 5.2. Deriving Procrastinate

The notion of derivational cost as proposed above straightforwardly

[10] This corresponds to the property of single-rootedness of phrase-markers in standard X'-Theory. Chomsky (1993:22) takes single-rootedness to be a convergence property at PF; Chomsky (1995:226), on the other hand, takes it to be a defining property of the mapping from N to λ. The shift is related to the fact that Chomsky (1995:chap. 4) allows lexical access in the covert component, but not in the phonological component (see fn. 3). It is reasonable to assume that in order for (the relevant features of) the lexical items shipped to the phonological component to be linearized in accordance with Kayne's (1994) Linear Correspondence Axiom, the syntactic object to be spelled out must also be single-rooted (but see Uriagereka forthcoming for the opposite view). The choice among these options is irrelevant for what follows.

accounts for the contrast between (3) and (4), repeated below in (19) and (20). After the structure in (21) is assembled, its strong feature can be checked by either lexical insertion (applications of Select and Merge) of *there* (cf. (19b)) or movement of *men* (cf. (20b)), both possibilities leading to a convergent derivation. Since Select and Merge are derivationally costles and Move is costly, lexical insertion is preferred over movement despite the fact that the former employs two operations.[11]

(19) a.There don't seem to be men in the list.
b. [ there$_i$ don't seem [ t$_i$ to be men in the list ] ]

(20) a. *There don't seem men to be in the list.
b. *[ there don't seem [ men$_i$ to be t$_i$ in the list ] ]

(21) [ to be men in the list ]

Similar reasoning extends to the derivational step after (6), repeated below in (22), is formed. Given that no strong feature requires that the Spec of Agro be filled, merging (22) with the tense head T is more economical than moving the object *Mary* to the Spec of Agro. The interesting question arises after the whole AgrsP in (2b), repeated below in (23), is formed with the movement of the subject *John* to the Spec of Agrs. Assuming for the moment that the numeration has already been exhausted (see below for further discussion), the sequence of derivational steps involving Select and Merge is no longer available. The next step then is either to apply the Spell-Out rule or to move *Mary* to the Spec of Agro for Case checking.

(22) [$_{\text{AgroP}}$ Agro [$_{\text{VP}}$ John saw Mary ] ]

(23) [$_{\text{AgrsP}}$ John [$_{\text{TP}}$ T [$_{\text{AgroP}}$ Agro [$_{\text{VP}}$ John saw Mary ] ] ] ]

[11] Nunes (1995:chap. IV) argues that rather than being a complex operation encompassing four suboperations (Copy, Merge, Form Chain, and Delete Trace), as in Chomsky's (1995:chap. 4) system, Move should be viewed as a description of the interaction of the independent operations Copy, Merge, Form Chain, and Chain Reduction (on the latter, see section 6 below). In this system, Copy is derivationally costly, but not Form Chain or Chain Reduction. Thus, "lexical insertion" (interaction of Select and Merge) is still more economical than overt movement (interaction of Copy, Merge, Form Chain, and Chain Reduction). See Nunes (1995:sec. II.10) for details.

According to what was discussed in the previous section, Spell-Out is a defining property of a derivation; if it does not apply, the pair (π, λ) is not formed and the derivation is cancelled. Spell-Out is therefore derivationally costless and should be preferred over movement, if the two options lead to convergence. In the case at hand, both options allow the derivation to converge, since strong features have already been checked. The computational system then applies Spell-Out to (23) and movement of (the formal features of) Mary takes place in the covert component.

Therefore, it is not be the case that overt movement is inherently more costly than covert movement, as stipulated by Procrastinate. Rather, this asymmetry follows from the fact that once strong features are checked and the numeration is exhausted, Spell-Out, which is derivationally costless, should be preferred over Move, which has derivational cost. The general economy considerations discussed in the previous section thus allow us to eliminate Procrastinate as a principle of UG, while deriving its effects without violating the Uniformity Condition on the mapping from N to λ. This analysis also overcomes the disadvantages of the proposals reviewed in section 4: economy is always computed locally, taking a single derivational step into account, and a unified account is offered to the preference for covert instead of overt movement, and the preference for lexical insertion instead of movement.[12]

For the sake of completeness, let us reconsider the sentence (2a),

[12] One reviewer raises the following issue:

ASSUME (i) that Spell-Out is taken to be part of the phonological component; and ASSUME further (ii) the view that such a system is extraneous to the computation $C_{HL}$. Then, Spell-Out (short of contradiction) CANNOT be taken to be "an operation which is a defining property of derivations", with the further consequence that it cannot even ENTER into comparison [in terms of cost; JNW]. If anything, Spell-Out should be a VERY costly operation, under this view.

I take Spell-Out to be an operation which is imposed on the computational system by what Chomsky's (1993, 1995) calls "bare output conditions". Given that (i) the language faculty interfaces with different cognitive systems (say, for the sake of the argument, the Conceptual-Intentional and the Articulatory-Perceptual systems) and (ii) these systems operate with different vocabularies, the computational system should have a screening device to satisfy the vocabulary requirements of each interface. Spell-Out fulfills such function by splitting the computation based on the kinds of lexical features each interface operates with. Spell-Out is therefore a defining property of syntactic derivations; if it does not apply, the pair (π,λ) can not be formed.

repeated below in (24a). Assuming that the "force" of a clause is determined by the nature of its complementizer, (24a) must have a null declarative complementizer C.[13] Thus, the structure in (24b) must be assembled at some point in the derivation. Based on the discussion in section 2, we conclude that the complementizer in (24b) does not have a strong feature, because it does not trigger overt movement. The question then is whether C is inserted overtly or covertly. Chomsky (1995:292) proposes that a complementizer with neither strong nor phonological features should be inserted covertly "on grounds of economy, if we assume that Procrastinate holds of Merge as well as Move".

(24) a. John saw Mary.
b.[$_{CP}$ C [$_{AgrsP}$ John [$_{TP}$ T [$_{AgroP}$ Agro [$_{VP}$ John saw Mary ] ] ] ] ]

Notice that underlying Chomsky's proposal is the assumption that Procrastinate should be taken as an independent economy principle of UG. If, on the other hand, the effects of Procrastinate should be derived along the lines proposed above, Chomsky's claim that Merge is subject to Procrastinate would amount to saying that Merge and Spell-Out should be compared for purposes of economy and that the former is more costly. However, there appears to be no principled reason for taking Merge to be inherently more costly than Spell-Out. In addition, the choice between overt and covert insertion of this type of complementizer seems to have no empirical consequence, as far as I can see. In absence of empirical evidence to the contrary, I will keep the assumption that Merge and Spell-Out should be analyzed as equally economical, given that a pair (π, λ) can only be formed with applications of these operations (see section 5.1).[14] It is possible that this is an instance in which the grammar allows true optionality: if Merge and Spell-Out are equally economical, a matrix complementizer with no

[13] As Chomsky (1995:292) observes, the null complementizer that appears in matrix clauses is different in nature from the overt complementizer *that* in English: the former carries declarative force, whereas the latter does not. Thus, (ii) is an appropriate answer for the question in (ia), but not for the one in (ib):

(i) a. What did Mary say?
b. What happened?
(ii) That John left.

[14] Recall that comparing Spell-Out with the sequence of derivational steps involving Select and Merge (i.e., lexical insertion) is not illuminating either, because Select is also derivationally costless (see section 5.1).

phonological or strong features can be inserted either before or after Spell-Out.

## 6. Other Economy Computations

In sections 6.2 and 6.3 below, I present two other cases which also fall under the conceptual guidelines for economy computation discussed in section 5.1. Both of them have to do with Nunes's 1995, 1996, forthcoming analysis of deletion of traces in the Minimalist Program, which is summarized in section 6.1.

### 6.1. The Copy Theory of Movement and Deletion of Traces

Assuming the general framework of Chomsky (1995:chap. 4), Nunes (1995, 1996, forthcoming) attempts to account for why traces must be deleted in the phonological component, once the copy theory of movement is assumed. Given the structure in (25) below, for instance, one must determine why the NP chain cannot be realized with all of its links phonetically realized (cf. (26a)) and why deletion targets traces and not the head of a chain (cf. (26b) vs. (26c)).

(25) [ John [ was [ arrested John ] ]

(26) a. *John was arrested John.
b. *Was arrested John.
c. John was arrested.

Extending a proposal by Chomsky (1995:227), Nunes (1995, 1996, forthcoming) assumes that two lexical items count as nondistinct if they are not distinctively specified in the initial numeration. In the case at hand, the two occurrences of *John* in (25) count as nondistinct if the initial numeration underlying (25) has a single instance of *John* (i.e., the index of *John* in the initial numeration is 1). Assuming this to be so, there is no way for the computational system to linearize the structure in (25) in accordance with Kayne's 1994 Linear Correspondence Axiom (LCA), according to which linear precedence in the phonological component is determined by asymmetric c-command. Since the verb *was* in (25), for instance, asymmetrically c-commands the lower instance of *John*, the LCA requires that *was* precede *John;* by the same token, the LCA requires that *John* precede *was* because the upper copy of *John* asymmetrically c-commands *was*. Given that the two copies of *John* are nondistinct, that amounts to saying that *was* should precede and be preceded by the same element, in violation of the asymmetry condition on linear order. Hence, the structure in (25) cannot surface as (26a) because it cannot be linearized. In order to yield a PF object, the NP-chain in (25) has to undergo the operation Chain Reduction, as described in (27) (see Nunes 1995, 1996, forthcoming for details and discussion).[15]

(27) *Chain Reduction:*
Delete the minimal number of constituents of a nontrivial chain CH which suffices for CH to be mapped into a linear order in accordance with the LCA.

Applying to (25), Chain Reduction deletes either the upper or the lower copy of *John*, allowing either resulting structure to be linearized in accordance with the LCA. The choice between these two derivations will depend on the elimination of formal features in the phonological component. Although formal features are relevant for morphological computations, they are not interpretable at PF (only phonological features are); thus, an operation of the phonological component applying after morphology must eliminate formal features which are visible at PF (see Chomsky, 1995:230-231). Let us refer to this rule as *FF-Elimination*, which is stated in (28) (see Nunes, 1995:291).

(28) *Formal Feature Elimination (FF-Elimination):*
Given the sequence of pairs $\sigma = <(F, P)_1, (F, P)_2, ..., (F, P)_n>$ such that $\sigma$ is the output of Linearize, F is a set of formal features and P is a set of phonological features, delete the minimal number of formal features in order for $\sigma$ to satisfy Full Interpretation at PF.

Extending Chomsky's 1995:sec. 4.5.2 checking theory, Nunes (1995) proposes that a [-interpretable] formal feature becomes invisible at PF after being checked. Thus, a checked feature need not (therefore must not) be eliminated by FF-Elimination, because it has already been rendered invisible

[15] Although I will assume the formulation in (27) for purposes of presentation, it is actually unnecessary to specify that Chain Reduction must delete the *minimal number* of constituents; that is, Chain Reduction need not count. Economy considerations regarding the length of a derivation may indirectly determine the number of elements to be deleted by enforcing the minimal number of applications of deletion. All things being equal, a short derivation should block a longer derivation (see Chomsky, 1995:314, 357); hence, a derivation in which constituents are unnecessarily deleted is longer, therefore less economical, than a competing derivation where no such deletion occurs. Similar considerations apply to FF-Elimination and Chain Uniformization, which are discussed below.

at PF by a checking operation (see Nunes 1995, 1996, forthcoming for details and discussion).

Bearing these considerations in mind, let us examine the Case-feature of *John* in the course of the derivation of (25), as shown in (29) below. The Case-feature of the upper copy of John becomes invisible at both LF and PF after being checked against the finite T head, as represented by the subscript in (29c).

(29) a. [ was [ arrested John-**CASE** ] ]
b. [ John-**CASE** [ was [ arrested John-**CASE** ] ] ]
c. [ John-$_{\text{CASE}}$ [ was [ arrested John-**CASE** ] ] ]

After (29c) undergoes Chain Reduction for purposes of linearization, it yields either (30a) or (30b) below, depending on which copy of *John* is deleted. In order to converge, the derivation operating with the structure in (30b) still requires an application of FF-Elimination targeting the unchecked Case-feature, whereas no such application in required for (30a), because its Case-feature became invisible at PF after being checked. The derivation in which

[16] Notice that the choice of the chain link to survive Chain Reduction is determined by *economy* considerations, not *convergence*. This makes the prediction that in instances where the phonetic realisation of the head of the chain does not lead to a convergent derivation, another link becomes the optimal option for phonetic realization. See Nunes forthcoming for discussion of potential cases.

[17] One reviewer asks whether this analysis does not wrongly predict that a structure such as (i), with movement of the expletive, should yield both sentences in (ii): given that the only formal feature of *there* (its categorial feature) enters into a checking relation with both the embedded and the matrix T, the two copies of *there* should be identical.

(i) [ there seems [ there to be a man in the room ] ]
(ii) a. There seems to be a man in the room.
b. *Seems there to be a man in the room.

Here I am following Nunes's (1995) proposal that when participating in an overt checking relation, a [+interpretable] feature can optionally be deleted with respect to PF. If it is deleted, it patterns with deleted [-interpretable] features in not being able to enter into any further checking relations; if it is not deleted with respect to PF, it is allowed to enter into another checking relation. Since undeleted formal features (regardless of their interpretability at the C-I interface) must be eliminated in the phonological component in order for the derivation to converge at PF, economy considerations dictate that two elements in an overt checking relation should have the largest number of features deleted with respect to PF, up to convergence. In other words, checking with respect to PF allows the number of applications of FF-Elimination targeting undeleted features to be minimized. Thus, if the D-feature of *there* (which I take to be [+interpretable], like any other categorial feature) is deleted with respect to PF in the embedded subject position in (i), it will not be able to check the strong feature of the matrix T; hence, only the upper copy of *there* in (i) can have its D-feature deleted for PF purposes, becoming the optimal link to survive Chain Reduction (cf. (iia)). Similar considerations extend to successive cyclic movement (see Nunes 1995:sec. III.6.2.5, forthcoming:sec. 6.1 for further details).

Chain Reduction deletes the head of the chain thus ends up being more costly than the one in which the trace is deleted; hence, the contrast between (26b) and (26c).[16,17]

(30) a. [ John-$_{\text{CASE}}$ [ was [ arrested ] ] ]
b.[ [ was [ arrested John-**CASE** ] ] ]

## 6.2. "Procrastinating" FF-Elimination

As formulated in (28), FF-Elimination applies after a given syntactic object is linearized and, therefore, after Chain Reduction has applied. This is crucial in the reasoning; if FF-Elimination applied to the NP-chain in (29c) before Chain Reduction, there would be no basis to distinguish (26b) from (26c), and, more generally, the account for why (in general) only heads of chains are phonetically realized would be lost.

Nunes (1995, 1996) observes that if FF-Elimination applied before Chain Reduction, it would be redundant in eliminating certain formal features of constituents which would be themselves deleted later on by Chain Reduction. Hence, application of Chain Reduction before FF-Elimination was taken to be the optimal option since it would avoid this redundancy. This reasoning faces the familiar problem of resorting to global economy computations since it takes into consideration two derivational steps at a time (see the discussion in section 4).

The conceptual grounds for economy considerations laid out in Chomsky (1995:226) and reviewed in section 5.1, however, provide the means for deriving the order of application between Chain Reduction and FF-Elimination in a local and unified fashion (see Nunes forthcoming). If the chain CH = (John-$_{\text{CASE}}$, John-**CASE**) in (29c), for instance, is not reduced, the structure containing it cannot be linearized and no PF object can be formed; as a defining property of a derivation, Chain Reduction is therefore costless. If FF-Elimination does not apply to (29c), on the other hand, an illegitimate PF object may eventually be formed; hence, by being associated with PF convergence, FF-Elimination is derivationally costly. Thus, in the derivational step where a chain can in principle undergo either Chain Reduction or FF-Elimination, economy considerations will ensure its reduction. Optimality considerations concerning the number of applications of FF-Elimination then indirectly choose the derivation where the lower links of the chain are deleted (see section 6.1).[18]

### 6.3. "Procrastinating" Chain Uniformization

Let us reconsider the structure in (29c), repeated below in (31). As is, (31) should yield a violation of Full Interpretation at LF because the Case-feature of the lower copy of *John*, a [-interpretable] feature, is visible at LF (see Chomsky, 1995:sec. 4.5.2).

(31) [ John-$_{\text{CASE}}$ [ was [ arrested John-**CASE** ] ] ]

The problem posed by (31) is reminiscent of the problem that a sentence such as (32a) below, for instance, presents for Chomsky's 1995 system, where "the features of a chain are considered a unit: if one is affected by an operation, all are" (see Chomsky 1995:chap. 4, fn. 12). Under this assumption, after the formal features of the lower copy of *what* in (32b) raise in the covert component, a checking operation will obliterate the Case-features of both links of the newly formed chain, but not the Case-feature of the copy of *what* in Spec of CP, which is part of the chain formed earlier in the overt syntax. Noting this problem, Chomsky (1995:303) further adds that "a convention is then needed requiring erasure of F throughout the array of chains containing F, so that no [-interpretable] feature remains in the operator position".

(32) a. What did John see?
b. [$_{\text{CP}}$ what-**CASE** did+Q [$_{\text{TP}}$ John see what-**CASE** ] ]

[18] Notice that this approach does not face the type of globality problem discussed in relation to Nunes (1994) and Kitahara (1995) (see section 4.1). In these papers, the application of an operation (overt movement) was contingent on the later application of another operation (deletion of traces). In the system explored here, Chain Reduction must apply regardless of FF-Elimination; the link to survive Chain Reduction is indirectly determined by economy considerations regarding derivational length (see fn. 15): the fewer features to be deleted by FF-Elimination a surviving link has, the shorter the derivation will be.

Assuming that traces are unaffected by the operations affecting heads of chains (see discussion in section 6.1), Nunes (1995) provides a single account of (31) and (32b) by implementing the convention suggested by Chomsky in terms of the condition in (33) and the operation in (34):

(33) *Feature Uniformity Condition:*
Given a chain CH = ($\alpha_1$, ..., $\alpha$n), every $\alpha$i ($1 \leq i \leq n$) must have the same set of features visible at LF.

(34) *Chain Uniformization:*
Delete the minimal number of features of a nontrivial chain CH in order to allow its links to satisfy the Feature Uniformity Condition.

As it stands, the NP chain in (31) violates the Feature Uniformity Condition in (33). Applied to (31), Chain Uniformization deletes the Case-feature of the lower copy of *John,* allowing the NP chain to satisfy the Feature Uniformity Condition and the derivation to converge at LF. As for (32a), we have to consider two chains: the chain $CH_1$ = (what-**CASE**, what-**CASE**), formed overtly, and the chain $CH_2$ = (FF(what-$_{CASE}$), FF(what-**CASE**)) formed after the set of formal features of the lower copy of *what* raises covertly. In order for $CH_2$ to satisfy the Feature Uniformity Condition, Chain Uniformization deletes the Case feature of its lower link, which consequently changes the uniform chain $CH_1$ into the nonuniform $CH_1$' = (what-**CASE**, what-$_{CASE}$). Chain Uniformization then applies to $CH_1$' and deletes the Case-feature of its upper link, allowing it to satisfy the Feature Uniformity Condition and the derivation to converge at LF.

Just to make sure that we do not have overapplications of (34), let us consider the chain CH = (Bill-**CASE**, Bill-**CASE**) in (35b) below. If Chain Uniformization deleted the unchecked Case-features of CH, the Feature Uniformity Condition would be satisfied, but the derivation in (35b) would be incorrectly allowed to converge, because Full Interpretation would be met. However, this incorrect result does not arise because Chain Uniformization does not apply to chains which are already uniform with

[19] This is actually the reason why Chain Uniformization cannot be subsumed under FF-Elimination, as one reviewer suggested: FF-Elimination is related to Full Interpretation (at PF), but Chain Uniformization is not.

respect to feature composition. The important thing to keep in mind is that, as stated in (34), deletion of ([-interpretable]) features is triggered by the Feature Uniformity Condition, not by Full Interpretation at LF. This is a natural assumption to make: if Chain Uniformization could delete any [-interpretable] feature to satisfy Full Interpretation at LF, no movement operation would ever be necessary.[19]

(35) a.*It was believed Bill to be often kissed.
b.[ it was believed [ Bill-**CASE** to [ be often kissed Bill-**CASE** ] ]

Let us now return to the issue of economy computations. I have been tacitly assuming that Chain Uniformization applies in the covert component. However, given that the Uniformity Condition on the mapping from a given numeration to LF makes the same set of operations available in the covert component and in overt syntax (see section 1), one wonders whether Chain Uniformization could apply to the chain of (36), for instance, before Spell-Out. If that were possible, it would enable the NP chain to satisfy Full Interpretation at both LF and PF without any other operation eliminating the unchecked Case-features; however, the basis for the NP trace to be deleted in the phonological component instead of the head of the chain would be lost (see section 6.1).

(36) [ John-$_{\text{CASE}}$ [ was [ arrested John-**CASE** ] ] ]

I propose that although available throughout the mapping from a given numeration to LF, Chain Uniformization is prevented from applying overtly for economy reasons. Consider a derivational step after all the strong features have been checked and the numeration has been exhausted. The computational system may then apply Chain Uniformization to the chains formed overtly or apply Spell-Out. Since Spell-Out is required for a derivation to be generated, it is costless, therefore being more economical than Chain Uniformization, which is an operation related to a convergence condition (the Feature Uniformity Condition). Thus, since the structure in (36) is spelled out without the uniformization of the NP chain, an asymmetry between the head and the tail is created, which will then be the basis for the choice of the link to be deleted in the phonological component (see section 6.1.). Therefore, the fact that Chain Uniformization only applies covertly need not be stipulated and is not at odds with the Uniformity Condition on the mapping from N to λ; its application after Spell-Out is ensured by general economy considerations which are independently motivated.

## 7. Conclusion

In a derivational view of the Minimalist Program, an adequate definition of what constitutes a possible derivation is obviously necessary. Chomsky (1995:225-226) proposes a definition and makes the interesting proposal that the operations of the computational system which are required in order for a given computation to be valid as a derivation so defined should be derivationally costless. The intuitive idea is that if these operations do not apply, we simply do not have a computation that is linguistically relevant; hence, it does not make sense to ask whether the resulting object is legitimate or whether a given computational step is more economical. Economy chooses among convergent derivations, therefore among derivations. Once the operations that are in some sense part of the definition of a possible derivation are taken to be costless, the remaining operations, the ones which are concerned with what is a legitimate LF or PF object, should thus be the ones which have derivational cost.

As observed by Chomsky, one of the effects of Procrastinate can be derived under this view: lexical insertion (applications of the costless operations Select and Merge) should always be preferred to overt movement (if the two options lead to convergent derivations). I have shown that the other aspect of Procrastinate (covert movement is more economical than overt movement) can also be derived along the same lines if we take the relevant comparison to be the one between Move and Spell-Out.

This approach has the virtue of stripping Procrastinate of any theoretical significance as a principle of economy. Recall that, as discussed in section 3, the Uniformity Condition on the mapping from N to $\lambda$ ensures that Spell-Out does not end up being a level of representation by being responsible for ruling out overt applications of "covert operations". By violating the Uniformity Condition, Procrastinate retained an unwanted residue of S-Structure in the system and therefore its effects should be accounted for in a different manner.

The notion of derivational cost depending on convergence was also shown to make the correct predictions with respect to the order of application of operations having to deal with deletion of traces as proposed in Nunes (1995. 1996. forthcoming). More specifically, (i) deletion of chain links for purposes of linearization (Chain Reduction) must precede elimination of formal features in the phonological component (FF-Elimination); and (ii) although Chain Uniformization (the operation which renders chains uniform in terms of feature composition) is available throughout the computation from N to λ. it only applies in the covert component.

To the extent that these results are derived in a unified fashion. they lend indirect support for Chomsky's (1995:225-226) definition of derivation as a sequence of symbolic elements S mapped from a numeration N such that the last member of S is a pair (π, λ) and N is reduced to zero.

(Recebido em julho de 1997. Aceito em outubro de 1997)

## References

Brody, M. (1995) *Lexico-Logical Form: a Radical Minimalist Theory.* Cambridge, Mass.: MIT Press.

Chomsky, N. (1981) *Lectures on Government and Binding.* Foris. Dordrecht.

_____ (1986) *Knowledge of Language: Its Nature, Origin and Use.* New York: Praeger.

_____ (1993) A Minimalist Program for Linguistic Theory. In: K. Hale & S. Keyser (eds.): *The Views from Building 20: Essays in Honor of Sylvain Bromberger.* 1-52. Cambridge, Mass.: MIT Press.

_____ (1994) Bare Phrase Structure. *MIT Occasional Papers 5.*

_____ (1995) *The Minimalist Program.* Cambridge, Mass.: MIT Press.

_____ Groat, E. & J. O'Neil (1996) Spell-Out at the Interface: Achieving a Unified Syntactic Computational System in the Minimalist Framework. In: W. Abraham. S. D. Epstein, H. Thráinsson & C. J.-W. Zwart (eds.): *Minimal Ideas: Syntactic Studies in the Minimalist Framework.* Amsterdam/Philadelphia: John Benjamins.

Kayne, R. (1994) *The Antisymmetry of Syntax.* Cambridge, Mass.: MIT Press.

Kitahara, H. (1995) Target α: Deducing Strict Cyclicity from Derivational Economy. *Linguistic Inquiry,* 26:46-77.

_____ (1997) *Elementary Operations and Optimal Derivations.* Cambridge. Mass.: MIT Press.

Nunes. J. (1994). Linearization of Non-trivial Chains at PF. *University of*

*Maryland Working Papers in Linguistics 2:159-177.*

_____ (1995) The Copy Theory of Movement and Linearization of Chains in the Minimalist Program. Doctoral dissertation. University of Maryland, College Park.

_____ (1996) On Why Traces Are Not Phonetically Realized. *Proceedings of NELS* 26, 211-225. Amherst: GLSA, University of Massachusetts. (forthcoming) Linearization of Chains and Phonetic Realization of Chain Links.

_____ To appear in: S. D. Epstein & N. Hornstein (eds.): *Working Minimalism.* Cambridge, Mass.: MIT Press.

_____ & E. THOMPSON (forthcoming) Appendix. In: J. Uriagereka: *Rhyme and Reason: a Minimalist Dialogue on Human Language.* Cambridge, Mass.: MIT Press.

URIAGEREKA, J. (1997) Multiple Spell-Out. *Groningen Arbeiten sur germanistischen Linguistik* 40:109-135.

_____ (forthcoming) *Rhyme and Reason: a Minimalist Dialogue on Human Language.* Cambridge, Mass.: MIT Press.